**IN THE CLAIMS:**

Please amend the claims as follows. (All claims listed)

1. (Original) A method of predicting instruction branches, comprising:

generating a current next-line prediction based on a previous next-line prediction; and

generating a current checking prediction based on the previous next-line prediction; and

generating a subsequent checking prediction based on the current next-line prediction, the checking predictions being independent from one another and having a longer latency than the next-line predictions.

2. (Original) The method of claim 1, further including:

comparing the current checking prediction to the current next-line prediction; and

updating a source of the next-line predictions based on the current checking prediction if the current next-line prediction does not have a target address that corresponds to a target address of the current checking prediction.

3. (Original) The method of claim 2, further including:

calculating a subset of the target address of the current checking prediction; and

comparing the subset to the target address of the current next-line prediction.

4. (Original) The method of claim 3, further including fetching one or more data blocks identified by the subset of the target address of the current checking prediction.

5. (Original) The method of claim 2, further including:

comparing the current checking prediction to an execution result; and

updating a source of the current checking prediction based on the execution result if the target address of the current checking prediction does not correspond to a target address of the execution result.

6. (Original) The method of claim 1, further including generating a subsequent next-line prediction based on the current next-line prediction.

7. (Original) The method of claim 1, wherein the next-line predictions are dynamic predictions.

8. (Original) The method of claim 1, wherein the previous next-line prediction has a latency that is a plurality of clock cycles and the previous next-line prediction includes a previous group prediction, the previous group prediction including a plurality of target addresses corresponding to the plurality of clock cycles.

9. (Original) The method of claim 8, wherein the plurality of target addresses includes a leaf target and one or more intermediate targets, the leaf target defining a target address of the group prediction.

10. (Original) The method of claim 9, further including:

hashing the leaf target to obtain an index; and

simultaneously indexing into a leaf array based on the index and into a block array based on the intermediate targets to obtain the current next-line prediction.

11. (Original) The method of claim 10, wherein the leaf array and the block array define a next-line prediction table.

12. (Original) The method of claim 8, further including generating a plurality of current checking predictions based on the plurality of target addresses, each of the plurality of current checking predictions being independent from one another.

13. (Original) The method of claim 12, further including generating a plurality of subsequent checking predictions based on the current next-line prediction, the plurality of current checking predictions being independent from the plurality of subsequent checking predictions, each of the plurality of subsequent checking predictions being independent from one another.

14. (Previously Presented) The method of claim 8, further including:

generating a bimodal prediction based on the previous next-line prediction;

generating a global prediction based on the previous next-line prediction;

generating a return prediction based on a return stack buffer value;

generating an indirect branch prediction based on an indirect branch value; and

selecting from the bimodal prediction, the global prediction, the return prediction and the indirect prediction to obtain the current next-line prediction.

15. (Original) A method of predicting instruction branches, comprising:

generating a current next-line prediction based on a previous next-line prediction, the previous next-line prediction having a latency that is a plurality of clock cycles, the previous next-line prediction including a previous group prediction, the previous group prediction including a plurality of target addresses corresponding to the plurality of clock cycles, the plurality of target addresses including a leaf target and one or more intermediate targets, the leaf target defining a target address of the previous prediction;

generating a plurality of current checking predictions based on the plurality of target addresses, each of the plurality of current checking predictions being independent from one another;

generating a plurality of subsequent checking predictions based on the current next-line prediction, the plurality of subsequent checking predictions being independent from the plurality of current checking predictions, each of the plurality of subsequent checking predictions being independent from one another, the next-line predictions being dynamic predictions.

16. (Original) The method of claim 15, further including:

hashing the leaf target to obtain an index; and

simultaneously indexing into a leaf array based on the index and into a block array based on the intermediate targets to obtain the current next-time predictions.

17. (Original) The method of claim 16, wherein the leaf array and the block array define a next-line predictor table.

18. (Original) A branch prediction architecture comprising:

a next-line predictor to generate a current next-line prediction based on a previous next-line prediction; and

a checking predictor to generate a current checking prediction based on the previous next-line prediction, and to generate a subsequent checking prediction based on the current next-line prediction, the checking predictions to be independent from one another and to have a longer latency than the next-line predictions.

19. (Original) The architecture of claim 18, further including a front end comparator to compare the current checking prediction to the current next-line prediction, and to update the next-line predictor based on the current checking prediction if the current next-line prediction does not have a target address that corresponds to a target address of the current checking prediction.

20. (Original) The architecture of claim 19, wherein the checking predictor is to calculate a subset of the target address of the current checking prediction, and to compare the subset to the target address of the current next-line prediction.

21. (Original) The architecture of claim 20, further including an instruction fetching unit to fetch one or more data blocks identified by the subset of the target address of the current checking prediction.

22. (Original) The architecture of claim 19, further including an execution comparator to compare the current checking prediction to an execution result, and to update the checking predictor based on the execution result if the target address of the current checking prediction does not correspond to a target address of the execution result.

23. (Original) The architecture of claim 18, wherein the next-line predictions are dynamic predictions.

24. (Original) The architecture of claim 18, wherein the previous next-line prediction is to have a latency that is a plurality of clock cycles and the previous next-line prediction is to include a previous group prediction, the previous group prediction to include a plurality of target addresses corresponding to the plurality of clock cycles.

25. (Original) The architecture of claim 24, wherein the plurality of target addresses is to include a leaf target and one or more intermediate targets, the leaf target to define a target address of the group prediction.

26. (Original) A computer system comprising:

a random access memory to store a branch instruction having an instruction address;

a system bus coupled to the memory; and

a processor coupled to the system bus, the processor having a next-line predictor and a checking predictor, the next-line predictor to generate a current next-line prediction based on the instruction address, the checking predictor to generate a current checking prediction based on the

instruction address, and to generate a subsequent checking prediction based on the current next-line prediction, the checking predictions to be independent from one another and to have a longer latency than the current next-line prediction.

27. (Original) The computer system of claim 26, further including a front end comparator to compare the current checking prediction to the current next-line prediction, and to update the next-line predictor based on the current checking prediction if the current next-line prediction does not have a target address that corresponds to a target address of the current checking prediction.

28. (Original) The computer system of claim 27, wherein the checking predictor is to calculate a subset of the target address of the current checking prediction, and to compare the subset to the target address of the current next-line prediction.

29. (Original) The computer system of claim 26, wherein the next-line prediction is dynamic.